Incremental Model-Checking of Delta-oriented Software Product Lines

FOSD Meeting 2015

ES Real-Time Systems Lab

Prof. Dr. rer. nat. Andy Schürr

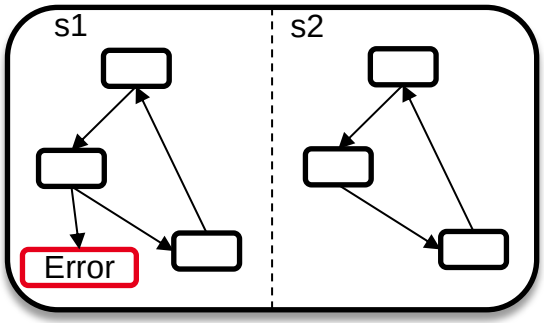
Dept. of Electrical Engineering and Information Technology

Dept. of Computer Science (adjunct Professor)

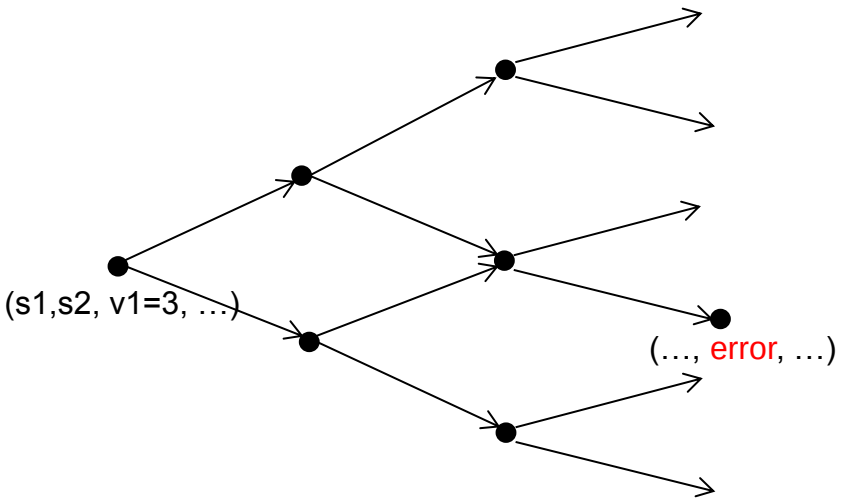
www.es.tu-darmstadt.de

Malte Lochau, Stephan Mennicke, Hauke Baller

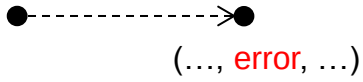
Model-Checking



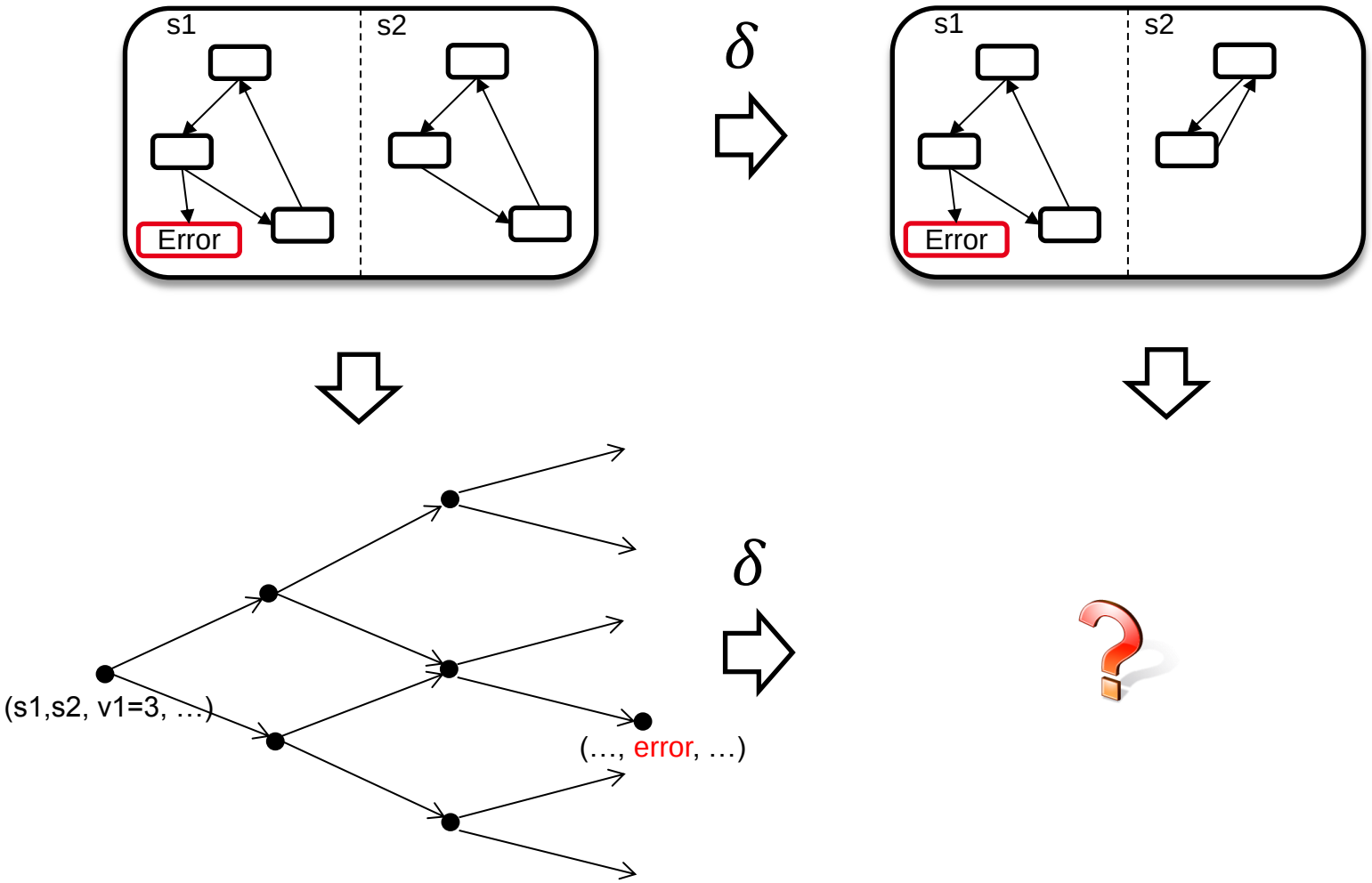
$\models$ $AG(\neg Error)$



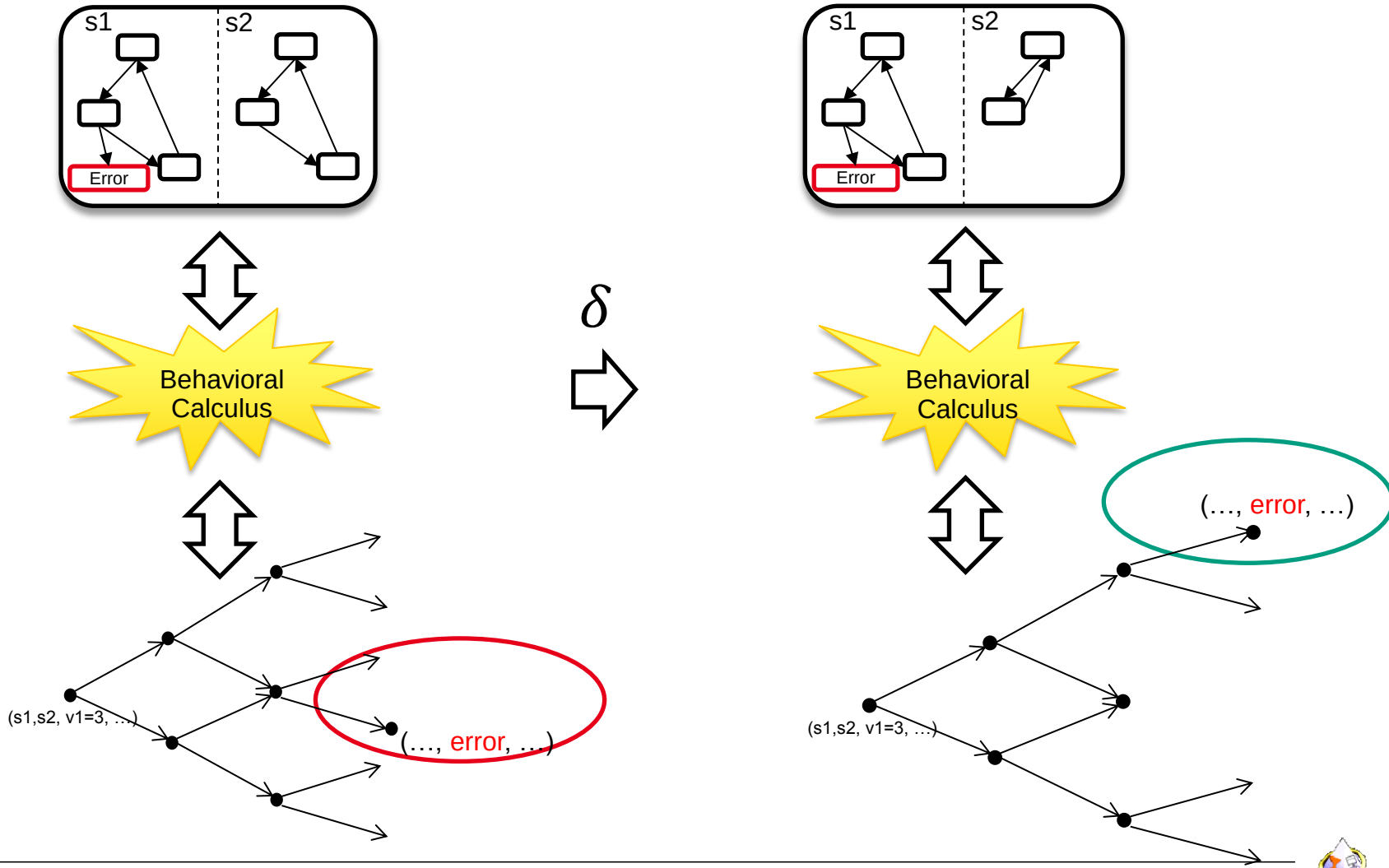
$\cap$



Syntactical vs. Semantical Changes



Semantical Models



A Short History of Behavioral Core Calculi

λ-Calculus
(1930)

CCS
(~1980)

CSP
(~1978)

λ- Calculus
+ Concurrency

CCS

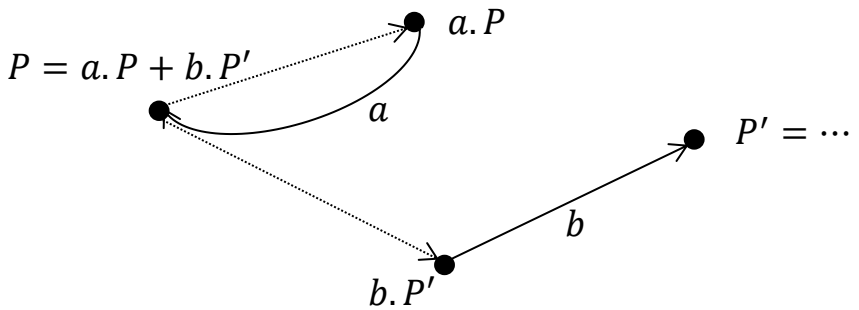
- CCS Syntax $P ::= \alpha.P \mid \sum_{i \in I} P_i \mid P \mid P \mid P[f] \mid P \setminus L \mid X$

- CCS Structural Operational Semantics

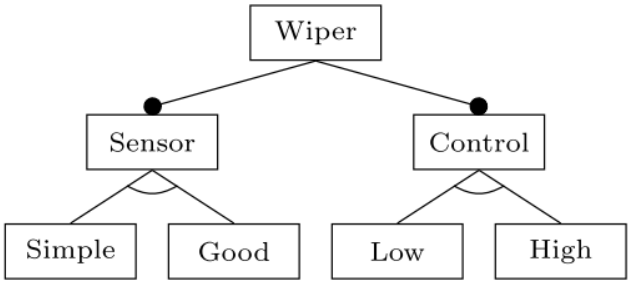
$$\text{(pre)}\ \frac{}{\alpha.P \xrightarrow{\alpha} P} \qquad \text{(rec)}\ \frac{P \xrightarrow{\alpha} P' \quad K \overset{Def}{=} P}{K \xrightarrow{\alpha} P'} \qquad \text{(choice)}\ \frac{P_j \xrightarrow{\alpha} P'_j \quad j \in I}{\sum_{i \in I} P_i \xrightarrow{\alpha} P'_j}$$

$$\text{(par-1)}\ \frac{P \xrightarrow{\alpha} P'}{P \mid Q \xrightarrow{\alpha} P' \mid Q} \qquad \text{(par-2)}\ \frac{Q \xrightarrow{\alpha} Q'}{P \mid Q \xrightarrow{\alpha} P \mid Q'} \qquad \text{(comm)}\ \frac{P \xrightarrow{\alpha} P' \quad Q \xrightarrow{\overline{\alpha}} Q'}{P \mid Q \xrightarrow{\tau} P' \mid Q'}$$

$$\text{(rel)}\ \frac{P \xrightarrow{\alpha} P'}{P[f] \xrightarrow{f(\alpha)} P'[f]} \qquad \text{(hide)}\ \frac{P \xrightarrow{\alpha} P' \quad \alpha, \overline{\alpha} \notin L}{P \setminus L \xrightarrow{\alpha} P' \setminus L}$$

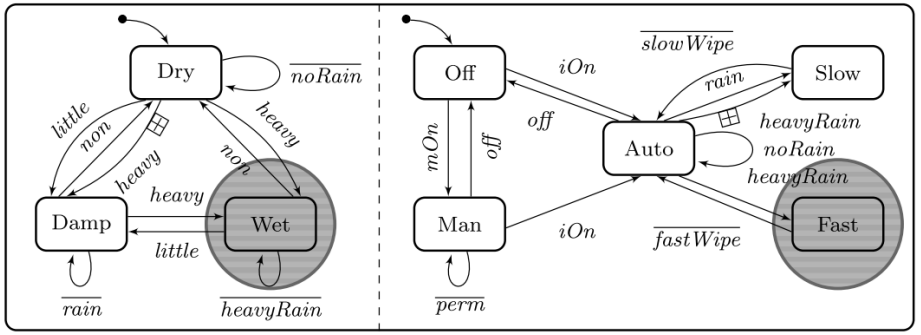


Delta-oriented Software Product Lines



Feature Model

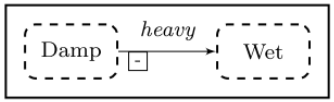
+



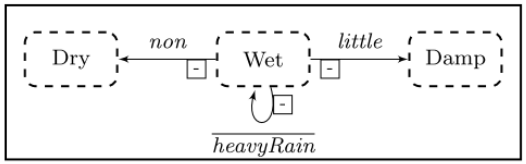
Core Product
(Good Sensor, High Control)

+

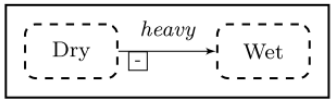
δ_1 $(\phi_1 := Simple)$:



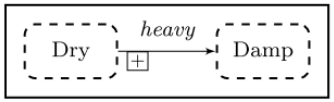
δ_2 $(\phi_1 := Simple)$:



δ_3 $(\phi_1 := Simple)$:

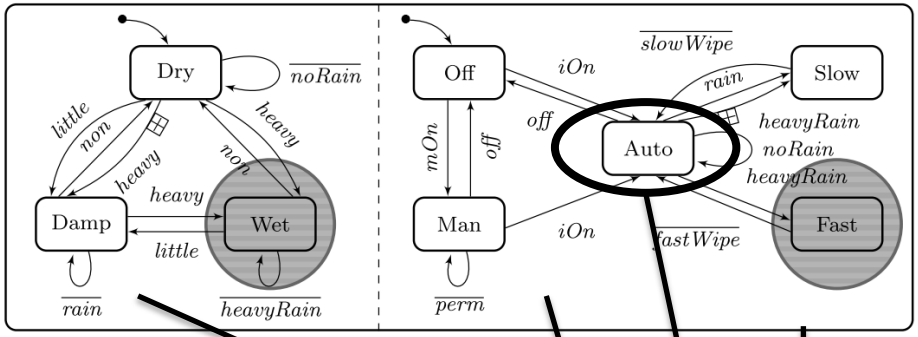


δ_4 $(\phi_1 := Simple)$:



Deltas
(Good Sensor => Simple Sensor)

Translation of State Machines into CCS



Initial process for root state with sub machines

$$P_c = \text{Dry} \mid \text{Off}$$

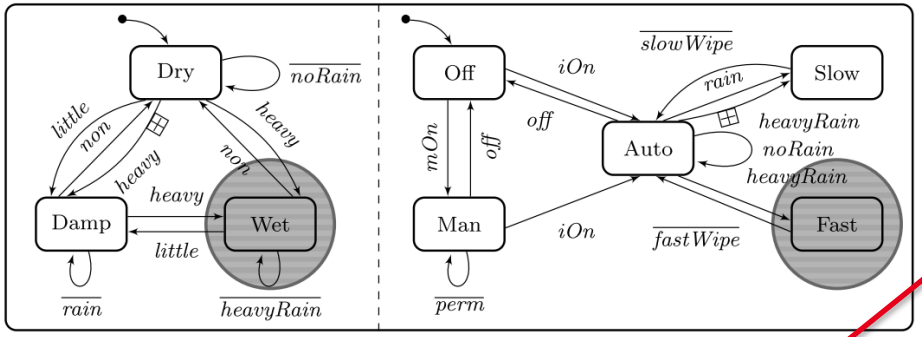
parallel sub processes for sub machines

$$\text{Dry} \overset{Def}{=} \overline{noRain}.\text{Dry} + little.\text{Damp} + heavy.\text{Wet}$$
$$\text{Damp} \overset{Def}{=} \overline{rain}.\text{Damp} + heavy.\text{Wet} + non.\text{Dry}$$
$$\text{Wet} \overset{Def}{=} \overline{heavyRain}.\text{Wet} + little.\text{Damp} + non.\text{Dry}$$

one process constant per state with choice over outgoing transitions

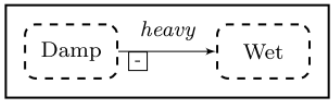
$$\text{Off} \overset{Def}{=} mOn.\text{Man} + iOn.\text{Auto}$$
$$\text{Man} \overset{Def}{=} \overline{perm}.\text{Man} + off.\text{Off} + ion.\text{Auto}$$
$$\text{Auto} \overset{Def}{=} noRain.\text{Auto} + rain.\text{Slow} + heavyRain.\text{Fast}$$
$$\text{Slow} \overset{Def}{=} \overline{slowWipe}.\text{Auto}$$
$$\text{Fast} \overset{Def}{=} \overline{fastWipe}.\text{Auto}$$

CCS Deltas

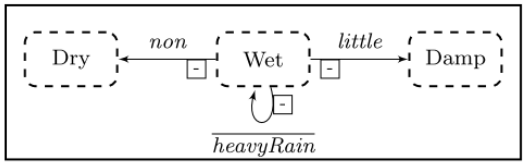


change transitions of state

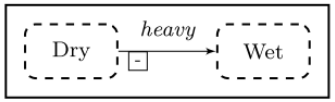
δ_1 $(\phi_1 := Simple)$:



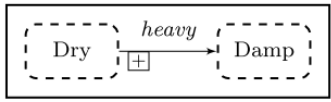
δ_2 $(\phi_1 := Simple)$:



δ_3 $(\phi_1 := Simple)$:



δ_4 $(\phi_1 := Simple)$:



$$\delta_1 = (\text{Damp}, \mathit{Simple}, \langle \text{Damp}' \overset{Def}{=} \overline{\mathit{rain}}.\text{Damp}' + \mathit{non}.\text{Dry} \rangle)$$

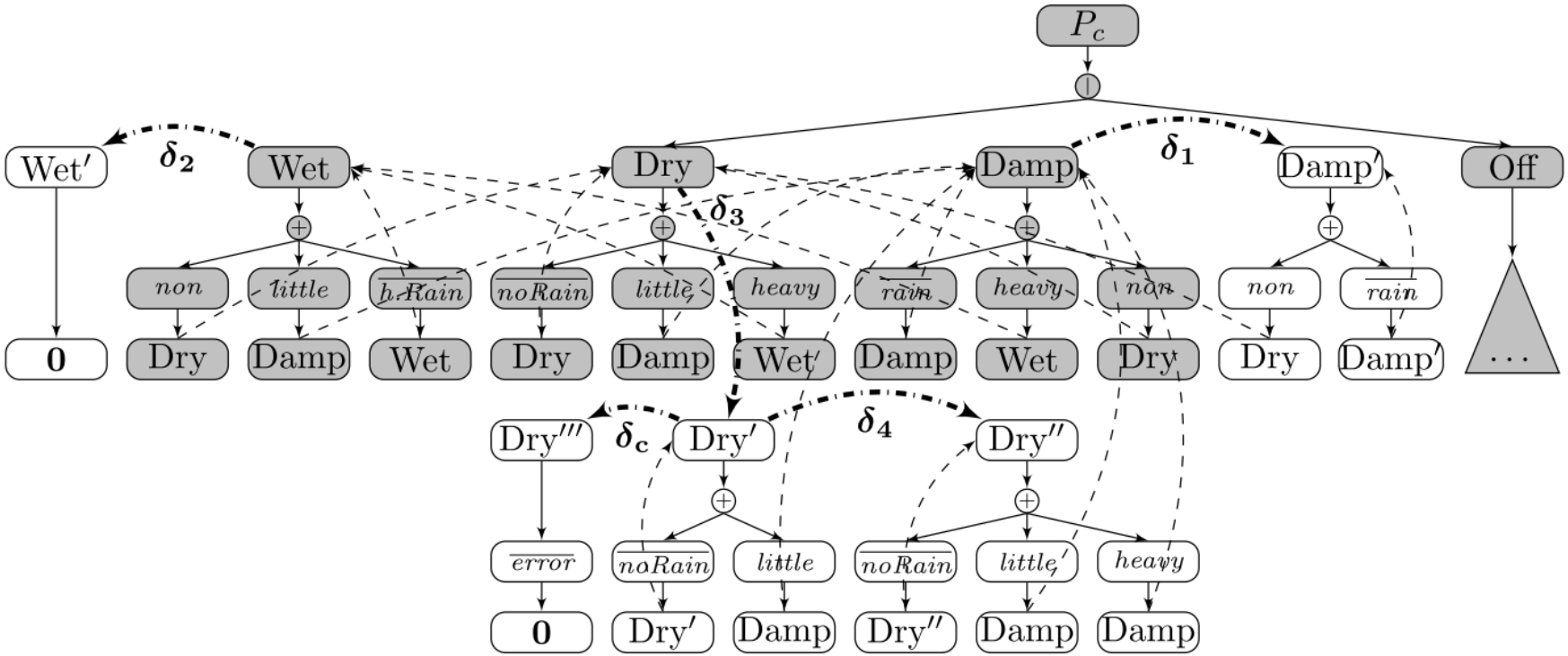
change process constant def. of state

$$\delta_2 = (\text{Wet}, \mathit{Simple}, \langle \text{Wet}' \overset{Def}{=} \mathbf{0} \rangle)$$

$$\delta_3 = (\text{Dry}, \mathit{Simple}, \langle \text{Dry}' \overset{Def}{=} \overline{\mathit{noRain}}.\text{Dry}' + \mathit{little}.\text{Damp} \rangle)$$

$$\delta_4 = (\text{Dry}', \mathit{Simple}, \langle \text{Dry}'' \overset{Def}{=} \overline{\mathit{noRain}}.\text{Dry}'' + \mathit{little}.\text{Damp} + \mathit{heavy}.\text{Damp} \rangle)$$

Delta Dependency Graph



Formalizing the Notion of (Correct) Behavior

TECHNISCHE UNIVERSITÄT DARMSTADT

$$P \models \varphi$$

- What is φ? temporal properties (safety, liveness)
- How to verify φ on CCS specifications P? model checking
- What happens with φ in $\delta(P)$? it depends…

What is φ?

TECHNISCHE UNIVERSITÄT DARMSTADT

Definition 4.1 (Modal μ-Calculus). *A* modal μ-calculus *formula is an expression following the form* $\nu Z.\psi \wedge [\alpha]Z$ *or* $\mu Z.\psi \vee \langle\alpha\rangle Z$ *where*

$$\psi \quad ::= \quad \mathit{tt} \mid \mathit{ff} \mid q \mid \neg q \mid Z \mid \psi \wedge \psi \mid \psi \vee \psi \mid \langle\alpha\rangle\psi \mid [\alpha]\psi$$

where $q \in \mathcal{P}$, $\alpha \in \mathsf{Act}$ *and* $Z \in \mathsf{Var}$. *Let* $P \in \mathcal{P}(\mathsf{Act}, \mathcal{K})$ *and* φ *a formula. Given an evaluation of atomic propositions* $\mathcal{I}_{\mathcal{P}} : \mathcal{P} \to 2^{\mathcal{P}(\mathsf{Act},\mathcal{K})}$ *and an evaluation of variables* $\mathcal{I}_{\mathsf{Var}} : \mathsf{Var} \to 2^{\mathcal{P}(\mathsf{Act},\mathcal{K})}$, $P \models \varphi$ *iff* $P \in \|\varphi\|^{\mathcal{I}_{\mathcal{P}}}_{\mathcal{I}_{\mathsf{Var}}}$ *(cf. Fig. 6).*

"[…] the system is intended to perform *fast wiping* whenever receiving the input *heavy… […]*":

$$\varphi := \mu Z.\langle heavy\rangle\langle\langle\overline{fastWipe}\rangle\rangle \mathtt{tt} \vee \langle -\rangle Z$$

What happens with φ in $\delta(P)$? (1/4)

TECHNISCHE UNIVERSITÄT DARMSTADT

(1) $P \simeq P'$ iff P and P' satisfy the same set of μ-calculus formulae

[Stirling et al.]

(2) $P \equiv P'$ implies $P \simeq P'$

[Milner et al.]

- Standard Congruence on CCS Terms:
  - $P + Q \equiv Q + P, P + P \equiv P, \ldots$
  - $P|Q \equiv Q|P, P|\mathbf{0} \equiv P$
  - $\alpha.(P + Q) \not\equiv \alpha.P + \alpha.Q$
  - ....

What happens with φ in $\delta(P)$? (2/4)

TECHNISCHE UNIVERSITÄT DARMSTADT

Proposition 4.1. *Let* $\delta = (K, \phi, K') \in \Delta(\mathcal{K}, \Phi)$ *and* $P, Q, P' \in \mathcal{P}(\mathsf{Act}, \mathcal{K})$.

$$\delta(\alpha.P) \equiv \alpha.\delta(P) \quad (1)$$

$$\delta(P + Q) \equiv \delta(P) + \delta(Q) \quad (2)$$

$$\delta(P \mid Q) \equiv \delta(P) \mid Q \text{ if } \delta(Q) \equiv Q \quad (3)$$

$$\delta(X) \equiv \delta(P') \text{ if } K \neq X \text{ and } X \stackrel{Def}{=} P' \quad (4)$$

$$\delta(P) \equiv P \text{ if } \delta \text{ is not applicable in } P \quad (5)$$

What happens with φ in $\delta(P)$? (4/4)

TECHNISCHE UNIVERSITÄT DARMSTADT

Proposition 4.2. *Let $P, Q, R \in \mathcal{P}(\mathsf{Act}, \mathcal{K})$ and φ a μ-calculus formula.*

$$P \models \varphi \wedge P \equiv Q \quad \Rightarrow \quad Q \models \varphi \tag{6}$$

$$P \models \varphi \wedge Q \models \varphi \quad \Rightarrow \quad P + Q \models \varphi \tag{7}$$

$$P \mid R \models \varphi \wedge P \equiv Q \quad \Rightarrow \quad Q \mid R \models \varphi \tag{8}$$

Theorem 4.1. *Let $\delta = (K, \phi, K') \in \Delta(\mathcal{K}, \Phi)$ and $P \in \mathcal{P}(\mathsf{Act}, \mathcal{K})$. If $\delta(P) \equiv P$ and $P \models \varphi$ then $\delta(P) \models \varphi$.*

[M. Lochau, S. Mennicke, H. Baller, L. Ribbeck: DeltaCCS: A Core Calculus for Behavioral Change, 2014]

Ongoing Work

TECHNISCHE UNIVERSITÄT DARMSTADT

- Incremental SPL model-checker in MAUDE
- Define normal forms for (theoretically) optimal congruence application strategies
- Explore second DeltaCCS semantics for family-based SPL model checking
- Interleave family-based and incremental SPL Test Suite Generation